UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/749,974	12/30/2003	Chih Ying Huang	MXICP014	9694

25920 7590 03/15/2007

MARTINE PENILLA & GENCARELLA, LLP
710 LAKEWAY DRIVE
SUITE 200
SUNNYVALE, CA 94085

EXAMINER	
ADAMS, CHARLES D	
ART UNIT	**PAPER NUMBER**
2164	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/15/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/749,974	HUANG ET AL.	
	Examiner	Art Unit	
	Charles D. Adams	2164	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *08 December 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-3,8-12 and 14-19* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-3,8-12 and 14-19* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Remarks

1. In response to communications filed on 8 December 2006, claims 1, 8-9, and 12 are amended and claims 4-7, 13 and 20-22 are cancelled. Claims 1-3, 8-12, and 14-19 are pending in the application.

Claim Objections

2. Claims 1 and 12 are objected to because of the following informalities: the claims contain a comma that renders the limitations unclear. For example, claim 1 has the limitation:

"Storing the received message in a file system included in the database server, when the received message includes data processed by the first processing machine"

Claim 12 contains the limitation:

"the daemon being further capable of storing the received message in a file system included in the database server, when the received message includes data processed by the first processing machine".

Claim Rejections - 35 USC § 103

3. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which

said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

4. Claims 1-3, 8-12, and 14-19 rejected under 35 U.S.C. 103(a) as being unpatentable over Merrick et al. (US Patent 7,028,312) in view of Manukyan (US Patent 7,177,897) and further in view of Birrell et al. ("Implementing Remote Procedure Calls").

As to claim 1, Merrick et al. teaches a method for data communication between processing machines (see Abstract):

Receiving a message produced by a first processing machine at a database server, the message being transmitted via socket protocol and received in a predetermined socket (see Merrick et al. 13:66-14:19 and 16:48-64. It is well known in the art that communications from using a certain program or protocol can arrive at predetermined ports. For example, Merrick et al. teaches "XML RPC may use any of a number of known transfer protocols, including network-specific protocols such as HTTP, SMPT, or FTP". Incoming SMTP messages arrive at TCP port 25);

Analyzing the received message in the database server (see 16:48-64);

Merrick et al. does not explicitly teach storing the received message in a file system included in the database server, when the received message includes data processed by the first processing machine;

Manukyan teaches storing the received message in a file system included in the database server, when the received message includes data processed by

the first processing machine (see 33:8-32. The inbound communication is stored);

Merrick et al. as modified teaches:

Transforming the received message into a database instruction when the received message is a simplified instruction, the transforming including dynamically identifying a structured query language (SQL) instruction from the simplified instruction (see Merrick et al. 16:48-64. An XML RPC instruction is transformed into an SQL instruction by extracting items from the XML RPC instruction);

Executing the SQL instruction to produce a result (see Merrick et al. 16:65-67); and

Sending the result to the first processing machine, wherein the message was produced as a result of processing within the first processing machine (see Merrick et al. 16:48-64)

Merrick et al. does not teach and the processing required the result to continue, and wherein the first processing machine receives a result set and resumes processing.

Birrell et al. teaches and the processing required the result to continue, and wherein the first processing machine receives a result set and resumes processing (see pages 39-40, section 1.1, background. "When the procedure finishes and produces its results, the results are passed back to the calling environment, where execution resumes as if returning from a simple single-machine call").

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have modified Merrick et al. by the teaching of Manukyan, since Manukyan teaches that "the frenetic pace of computer innovation has increased the need for hosting companies to provide services that are quickly accessible and have enhanced performance. For instance, many end users utilize the Web to transact business, order supplies, and exchange information. As a result, hosting companies are under increased pressure to deliver hosting services to clients that are more accessible, problem free, and match the rapid pace in which services are utilized by the end-users or customers of the client" (see 1:41-50).

Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to have modified Merrick et al. by the teaching of Birrell et al., since Birrell et al. teaches "the idea of remote procedure calls is quite simple. It is based on the observation that procedure calls are a well-known and well-understood mechanism for transfer of control and data within a program running on a single computer. Therefore, it is proposed that this same mechanism be extended to provide for transfer of control and data across a communication network" (see pages 39-40, section 1.1, background). In addition to this, Merrick et al. teaches using remote procedure calls, while Birrell et al. simply goes into more depth about the functions of remote procedure calls.

As to claim 2, Merrick et al. as modified teaches wherein the message produced by the first processing machine includes a header formatted to identify whether the message is a database instruction (see Merrick et al. 17:32-56).

As to claim 3, Merrick et al. as modified teaches wherein analyzing the received message in the database server includes analyzing the header of the message to determine if the message includes the database instruction (see Merrick et al. 16:48-64 and 17:32-56).

As to claim 8, Merrick et al. teaches wherein the database server includes a daemon and wherein transforming the received message includes transforming the received message in the daemon (see Merrick et al. 16:48-64, 18:29-39. A daemon is a program that waits inactive until its functionality is needed. The programs in Merrick et al. are functionally equivalent to this in that they respond when they receive a message).

As to claim 9, Merrick et al. as modified teaches wherein the daemon monitors the predetermined socket for messages from the first processing machine (see Manukyan 33:8-32).

As to claim 10, Merrick et al. teaches wherein the message produced by the first processing machine is transmitted to the database server as data included in the message that is produced (see Merrick et al. 16:48-64, 17:32-56).

As to claim 11, Merrick et al. teaches wherein the message being transmitted via socket protocol includes being transmitted via TCP/IP protocol (see Merrick et al. 13:66-14:19).

As to claim 12, Merrick et al. teaches a system for communicating data between processing machines comprising:

A database server (see 13:66-14:19);

At least one processing machine capable of communicating with the database server via a socket protocol and via a predetermined socket (see 13:66-14:19);

A network for coupling the at least one processing machine to the database server (see Merrick et al. 13:66-14:19 and 16:48-64. It is well known in the art that communications from a certain program will arrive at predetermined ports. For example, Merrick et al. teaches "XML RPC may use any of a number of known transfer protocols, including network-specific protocols such as HTTP, SMPT, or FTP". Incoming SMTP messages always arrive at a certain TCP port (namely, 25); and

a daemon included within the database server (see 16:48-64 and 18:29-39),

Merrick et al. does not teach explicitly teach the daemon being capable of monitoring the predetermined socket for a message from a first processing machine of the at least one processing machine, and the daemon being further

capable of storing the received message in a file system included in the database server, when the received message includes data processed by the first processing machine,

Manukyan teaches the daemon being capable of monitoring the predetermined socket for a message from a first processing machine of the at least one processing machine, and the daemon being further capable of storing the received message in a file system included in the database server, when the received message includes data processed by the first processing machine (see 33:8-32. The inbound communication is stored),

Merrick et al. as modified teaches:

Determining if a message received from the at least one processing machine is a simplified database instruction, transforming the received message into a database instruction when the received message is a simplified instruction, the transforming including dynamically identifying a structured query language (SQL) instruction from the simplified instruction (see Merrick et al. 16:48-64. An XML RPC instruction is transformed into an SQL instruction by extracting items from the XML RPC instruction),

Executing the SQL instruction to produce a result, and sending the result to the first processing machine (see Merrick et al. 16:48-67),

Merrick et al. does explicitly teach wherein the message was produced as a result of processing within the first processing machine and the processing required the result to continue, and wherein the first processing machine receives a result set and resumes processing

Birrell et al. teaches wherein the message was produced as a result of processing within the first processing machine and the processing required the result to continue, and wherein the first processing machine receives a result set and resumes processing (see pages 39-40, section 1.1, background. "When the procedure finishes and produces its results, the results are passed back to the calling environment, where execution resumes as if returning from a simple single-machine call")..

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have modified Merrick et al. by the teaching of Manukyan, since Manukyan teaches that "the frenetic pace of computer innovation has increased the need for hosting companies to provide services that are quickly accessible and have enhanced performance. For instance, many end users utilize the Web to transact business, order supplies, and exchange information. As a result, hosting companies are under increased pressure to deliver hosting services to clients that are more accessible, problem free, and match the rapid pace in which services are utilized by the end-users or customers of the client" (see 1:41-50).

Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to have modified Merrick et al. by the teaching of Birrell et al., since Birrell et al. teaches "the idea of remote procedure calls is quite simple. It is based on the observation that procedure calls are a well-known and well-understood mechanism for transfer of control and data within a program running on a single computer. Therefore, it is proposed that this same

mechanism be extended to provide for transfer of control and data across a communication network" (see pages 39-40, section 1.1, background). In addition to this, Merrick et al. teaches using remote procedure calls, while Birrell et al. simply goes into more depth about the functions of remote procedure calls.

As to claim 14, Merrick et al. teaches wherein each of the at least one processing machine includes a corresponding operating system (see 13:66-14:19. It is inherent that a functional computer have an operating system).

As to claim 15, Merrick et al. teaches wherein the corresponding operating systems (see 13:66-14:19. It is inherent that a functional computer have an operating system).

The remainder of the claim is optionally recited, and thus bears no patentable weight.

As to claim 16, Merrick et al. teaches wherein a header of the received message includes an identification of the message as including the database instruction (see 16:48-64 and 17:32-56).

As to claim 17, Merrick et al. teaches wherein the daemon executes dynamic language compiling (see 16:48-64).

As to claim 18, Merrick et al. teaches wherein the dynamic language compiling includes SQL language compiling that has the capability of converting the message to an SQL instruction (see 16:48-64).

As to claim 19, Merrick et al. teaches wherein the at least one processing machine is capable of communicating with the database server via a TCP/IP protocol (see 13:66-14:19).

Response to Arguments

5. Applicant's arguments filed 8 December 2006 have been fully considered but they are not persuasive.

Applicant argues that "the use of an XML-based message as shown in Merrick et al. is not the same as the transforming of the received message into a database instruction when the received message is a simplified instruction as specified in the presently claimed subject matter". In response to this argument, Examiner notes that the XML instruction is a simplified instruction (see 13:57-65. XML is self-describing), and the instruction is transformed by changing the simplified XML RPC call to an SQL query by extracting arguments from the XML instruction.

Applicant argues that Merrick et al. does not teach a daemon in 16:48-64 and 18:29-39. While the word 'daemon' is not explicitly mentioned, one of

ordinary skill in the art knows that a 'daemon' is a program that runs continuously and handles service requests as they arrive. It runs in the background and performs functionality when it is required. As Merrick et al. teaches that the message is identified and converted upon arrival, it is functionally equivalent to the teaching of a 'daemon'.

Conclusion

6. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Charles D. Adams whose telephone number is (571) 272-3938. The examiner can normally be reached on 8:30 AM - 5:00 PM, M - F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Charles Rones can be reached on (571) 272-4085. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Charles Adams
AU2164

CA

Primary Examiner
Cam Y Truong